

Práctica 2.1: Introducción a la programación de sistemas UNIX

Objetivos

En esta práctica estudiaremos el uso básico del API de un sistema UNIX y su entorno de desarrollo. En particular, se usarán funciones para gestionar errores y obtener información.

Contenidos

- Preparación del entorno para la práctica
- Gestión de errores
- Información del sistema
- Información del usuario
- Información horaria del sistema

Preparación del entorno para la práctica

Esta práctica únicamente requiere el entorno de desarrollo (compilador, editores y depurador), que está disponible en las máquinas virtuales de la asignatura y en la máquina física del laboratorio.

Se puede usar cualquier editor gráfico o de terminal. Además, se puede usar tanto el lenguaje C (compilador gcc) como C++ (compilador g++). Si fuera necesario compilar varios archivos, se recomienda el uso de make. Finalmente, el depurador recomendado en las prácticas es gdb. **No está permitido** el uso de IDEs como Eclipse.

Gestión de errores

Usar las funciones disponibles en el API del sistema (perror(3) y strerror(3)) para gestionar los errores en los siguientes casos. En cada ejercicio, añadir las librerías necesarias (#include).

Ejercicio 1. Añadir el código necesario para gestionar correctamente los errores generados por la llamada a setuid(2). Consultar en el manual el propósito de la llamada y su prototipo.

```
#include <sys/types.h>
#include <unistd.h>
int setuid(uid_t uid); el prototipo

#include <sys/types.h>
#include <unistd.h>
#include <errno.h>
#include <stdio.h>
#include <string.h>
int main() {
    int ret= setuid(0);
    if (ret==-1){
        perror("error en el setuid");
        return 0;
    }
    return 1;
}
```

```
g++ pr1.cpp -o pr1
./pr1
Error en el setuid: Operation not permitted
```

Ejercicio 2. Imprimir el código de error generado por la llamada del código anterior, tanto en su versión numérica como la cadena asociada.

```
int main() {
    int ret= setuid(0);
    printf ("Codigo del error : %i \n", errno);
    printf ("Cadena asociada al error : %s \n", strerror(errno));
    return 1;
}
gcc pr12.c -o pr2
./pr2
Código del error : 1
Cadena asociada al error : Operation not permitted
```

Ejercicio 3. Escribir un programa que imprima todos los mensajes de error disponibles en el sistema. Considerar inicialmente que el límite de errores posibles es 255.

```
int main() {
    int i=0;
    for(i=0;i<255;i++)
    {
        printf ("Codigo del error : %i. Cadena asociada al error : %s \n",i,
strerror(i));
    }
    return 1;
}

gcc pr13.c -o pr3
./pr3
Codigo del error : 0. Cadena asociada al error : Success
Codigo del error : 1. Cadena asociada al error : Operation not permitted
Codigo del error : 2. Cadena asociada al error : No such file or directory
Codigo del error : 3. Cadena asociada al error : No such process
Codigo del error : 4. Cadena asociada al error : Interrupted system call
Codigo del error : 5. Cadena asociada al error : Input/output error
Codigo del error : 6. Cadena asociada al error : No such device or address
Codigo del error : 7. Cadena asociada al error : Argument list too long
Codigo del error : 8. Cadena asociada al error : Exec format error
Codigo del error : 9. Cadena asociada al error : Bad file descriptor
Codigo del error : 10. Cadena asociada al error : No child processes
Codigo del error : 11. Cadena asociada al error : Resource temporarily
unavailable
Codigo del error : 12. Cadena asociada al error : Cannot allocate memory
Codigo del error : 13. Cadena asociada al error : Permission denied
```

Codigo del error : 14. Cadena asociada al error : Bad address
 Codigo del error : 15. Cadena asociada al error : Block device required
 Codigo del error : 16. Cadena asociada al error : Device or resource busy
 Codigo del error : 17. Cadena asociada al error : File exists
 Codigo del error : 18. Cadena asociada al error : Invalid cross-device link
 Codigo del error : 19. Cadena asociada al error : No such device
 Codigo del error : 20. Cadena asociada al error : Not a directory
 Codigo del error : 21. Cadena asociada al error : Is a directory
 Codigo del error : 22. Cadena asociada al error : Invalid argument
 Codigo del error : 23. Cadena asociada al error : Too many open files in system
 Codigo del error : 24. Cadena asociada al error : Too many open files
 Codigo del error : 25. Cadena asociada al error : Inappropriate ioctl for device
 Codigo del error : 26. Cadena asociada al error : Text file busy
 Codigo del error : 27. Cadena asociada al error : File too large
 Codigo del error : 28. Cadena asociada al error : No space left on device
 Codigo del error : 29. Cadena asociada al error : Illegal seek
 Codigo del error : 30. Cadena asociada al error : Read-only file system
 Codigo del error : 31. Cadena asociada al error : Too many links
 Codigo del error : 32. Cadena asociada al error : Broken pipe
 Codigo del error : 33. Cadena asociada al error : Numerical argument out of domain
 Codigo del error : 34. Cadena asociada al error : Numerical result out of range
 Codigo del error : 35. Cadena asociada al error : Resource deadlock avoided
 Codigo del error : 36. Cadena asociada al error : File name too long
 Codigo del error : 37. Cadena asociada al error : No locks available
 Codigo del error : 38. Cadena asociada al error : Function not implemented
 Codigo del error : 39. Cadena asociada al error : Directory not empty
 Codigo del error : 40. Cadena asociada al error : Too many levels of symbolic links
 Codigo del error : 41. Cadena asociada al error : Unknown error 41
 Codigo del error : 42. Cadena asociada al error : No message of desired type
 Codigo del error : 43. Cadena asociada al error : Identifier removed
 Codigo del error : 44. Cadena asociada al error : Channel number out of range
 Codigo del error : 45. Cadena asociada al error : Level 2 not synchronized
 Codigo del error : 46. Cadena asociada al error : Level 3 halted
 Codigo del error : 47. Cadena asociada al error : Level 3 reset
 Codigo del error : 48. Cadena asociada al error : Link number out of range
 Codigo del error : 49. Cadena asociada al error : Protocol driver not attached
 Codigo del error : 50. Cadena asociada al error : No CSI structure available
 Codigo del error : 51. Cadena asociada al error : Level 2 halted
 Codigo del error : 52. Cadena asociada al error : Invalid exchange
 Codigo del error : 53. Cadena asociada al error : Invalid request descriptor
 Codigo del error : 54. Cadena asociada al error : Exchange full
 Codigo del error : 55. Cadena asociada al error : No anode
 Codigo del error : 56. Cadena asociada al error : Invalid request code
 Codigo del error : 57. Cadena asociada al error : Invalid slot
 Codigo del error : 58. Cadena asociada al error : Unknown error 58
 Codigo del error : 59. Cadena asociada al error : Bad font file format
 Codigo del error : 60. Cadena asociada al error : Device not a stream
 Codigo del error : 61. Cadena asociada al error : No data available
 Codigo del error : 62. Cadena asociada al error : Timer expired

Codigo del error : 63. Cadena asociada al error : Out of streams resources
 Codigo del error : 64. Cadena asociada al error : Machine is not on the network
 Codigo del error : 65. Cadena asociada al error : Package not installed
 Codigo del error : 66. Cadena asociada al error : Object is remote
 Codigo del error : 67. Cadena asociada al error : Link has been severed
 Codigo del error : 68. Cadena asociada al error : Advertise error
 Codigo del error : 69. Cadena asociada al error : Srmount error
 Codigo del error : 70. Cadena asociada al error : Communication error on send
 Codigo del error : 71. Cadena asociada al error : Protocol error
 Codigo del error : 72. Cadena asociada al error : Multihop attempted
 Codigo del error : 73. Cadena asociada al error : RFS specific error
 Codigo del error : 74. Cadena asociada al error : Bad message
 Codigo del error : 75. Cadena asociada al error : Value too large for defined data type
 Codigo del error : 76. Cadena asociada al error : Name not unique on network
 Codigo del error : 77. Cadena asociada al error : File descriptor in bad state
 Codigo del error : 78. Cadena asociada al error : Remote address changed
 Codigo del error : 79. Cadena asociada al error : Can not access a needed shared library
 Codigo del error : 80. Cadena asociada al error : Accessing a corrupted shared library
 Codigo del error : 81. Cadena asociada al error : .lib section in a.out corrupted
 Codigo del error : 82. Cadena asociada al error : Attempting to link in too many shared libraries
 Codigo del error : 83. Cadena asociada al error : Cannot exec a shared library directly
 Codigo del error : 84. Cadena asociada al error : Invalid or incomplete multibyte or wide character
 Codigo del error : 85. Cadena asociada al error : Interrupted system call should be restarted
 Codigo del error : 86. Cadena asociada al error : Streams pipe error
 Codigo del error : 87. Cadena asociada al error : Too many users
 Codigo del error : 88. Cadena asociada al error : Socket operation on non-socket
 Codigo del error : 89. Cadena asociada al error : Destination address required
 Codigo del error : 90. Cadena asociada al error : Message too long
 Codigo del error : 91. Cadena asociada al error : Protocol wrong type for socket
 Codigo del error : 92. Cadena asociada al error : Protocol not available
 Codigo del error : 93. Cadena asociada al error : Protocol not supported
 Codigo del error : 94. Cadena asociada al error : Socket type not supported
 Codigo del error : 95. Cadena asociada al error : Operation not supported
 Codigo del error : 96. Cadena asociada al error : Protocol family not supported
 Codigo del error : 97. Cadena asociada al error : Address family not supported by protocol
 Codigo del error : 98. Cadena asociada al error : Address already in use
 Codigo del error : 99. Cadena asociada al error : Cannot assign requested address
 Codigo del error : 100. Cadena asociada al error : Network is down
 Codigo del error : 101. Cadena asociada al error : Network is unreachable
 Codigo del error : 102. Cadena asociada al error : Network dropped connection on reset
 Codigo del error : 103. Cadena asociada al error : Software caused connection abort

Codigo del error : 104. Cadena asociada al error : Connection reset by peer
 Codigo del error : 105. Cadena asociada al error : No buffer space available
 Codigo del error : 106. Cadena asociada al error : Transport endpoint is already connected
 Codigo del error : 107. Cadena asociada al error : Transport endpoint is not connected
 Codigo del error : 108. Cadena asociada al error : Cannot send after transport endpoint shutdown
 Codigo del error : 109. Cadena asociada al error : Too many references: cannot splice
 Codigo del error : 110. Cadena asociada al error : Connection timed out
 Codigo del error : 111. Cadena asociada al error : Connection refused
 Codigo del error : 112. Cadena asociada al error : Host is down
 Codigo del error : 113. Cadena asociada al error : No route to host
 Codigo del error : 114. Cadena asociada al error : Operation already in progress
 Codigo del error : 115. Cadena asociada al error : Operation now in progress
 Codigo del error : 116. Cadena asociada al error : Stale file handle
 Codigo del error : 117. Cadena asociada al error : Structure needs cleaning
 Codigo del error : 118. Cadena asociada al error : Not a XENIX named type file
 Codigo del error : 119. Cadena asociada al error : No XENIX semaphores available
 Codigo del error : 120. Cadena asociada al error : Is a named type file
 Codigo del error : 121. Cadena asociada al error : Remote I/O error
 Codigo del error : 122. Cadena asociada al error : Disk quota exceeded
 Codigo del error : 123. Cadena asociada al error : No medium found
 Codigo del error : 124. Cadena asociada al error : Wrong medium type
 Codigo del error : 125. Cadena asociada al error : Operation canceled
 Codigo del error : 126. Cadena asociada al error : Required key not available
 Codigo del error : 127. Cadena asociada al error : Key has expired
 Codigo del error : 128. Cadena asociada al error : Key has been revoked
 Codigo del error : 129. Cadena asociada al error : Key was rejected by service
 Codigo del error : 130. Cadena asociada al error : Owner died
 Codigo del error : 131. Cadena asociada al error : State not recoverable
 Codigo del error : 132. Cadena asociada al error : Operation not possible due to RF-kill
 Codigo del error : 133. Cadena asociada al error : Memory page has hardware error
 Codigo del error : 134. Cadena asociada al error : Unknown error 134
 Codigo del error : 135. Cadena asociada al error : Unknown error 135
 Codigo del error : 136. Cadena asociada al error : Unknown error 136
 Codigo del error : 137. Cadena asociada al error : Unknown error 137
 Codigo del error : 138. Cadena asociada al error : Unknown error 138
 Codigo del error : 139. Cadena asociada al error : Unknown error 139
 Codigo del error : 140. Cadena asociada al error : Unknown error 140
 Codigo del error : 141. Cadena asociada al error : Unknown error 141
 Codigo del error : 142. Cadena asociada al error : Unknown error 142
 Codigo del error : 143. Cadena asociada al error : Unknown error 143
 Codigo del error : 144. Cadena asociada al error : Unknown error 144
 Codigo del error : 145. Cadena asociada al error : Unknown error 145
 Codigo del error : 146. Cadena asociada al error : Unknown error 146
 Codigo del error : 147. Cadena asociada al error : Unknown error 147
 Codigo del error : 148. Cadena asociada al error : Unknown error 148
 Codigo del error : 149. Cadena asociada al error : Unknown error 149

Ampliación de Sistemas Operativos y Redes

Ampliación de Sistemas Operativos y Redes


```
Codigo del error : 252. Cadena asociada al error : Unknown error 252
Codigo del error : 253. Cadena asociada al error : Unknown error 253
Codigo del error : 254. Cadena asociada al error : Unknown error 254
```

Información del sistema

Ejercicio 4. El comando del sistema `uname(1)` muestra información sobre diversos aspectos del sistema. Consultar la página de manual y obtener la información del sistema.

```
uname --all
Linux pto0811 5.8.0-63-generic #71~20.04.1-Ubuntu SMP Thu Jul 15 17:46:08 UTC
2021 x86_64 x86_64 x86_64 GNU/Linux
```

Ejercicio 5. Escribir un programa que muestre, con `uname(2)`, cada aspecto del sistema y su valor. Comprobar la correcta ejecución de la llamada.

```
#include <sys/types.h>
#include <unistd.h>
#include <errno.h>
#include <stdio.h>
#include <string.h>
#include <sys/utsname.h>

int main() {
    struct utsname s;
    if(uname(&s) == -1){//EFAULT
        perror("fallos en UNAME");
        return 1;
    }

    printf("sysname: %s\n",s.sysname);
    printf("nodename: %s\n",s.nodename);
    printf("release: %s\n",s.release);
    printf("version: %s\n",s.version);
    printf("machine: %s\n",s.machine);
    printf("domainname: %s\n",s.domainname);
    return 0;
}
```

```
./pr5
sysname: Linux
nodename: localhost.localdomain
release: 3.10.0-862.11.6.el7.x86_64
version: #1 SMP Tue Aug 14 21:49:04 UTC 2018
machine: x86_64
domainname: (none)
```


Ejercicio 6. Escribir un programa que obtenga, con `sysconf(3)`, información de configuración del sistema e imprima, por ejemplo, la longitud máxima de los argumentos, el número máximo de hijos y el número máximo de ficheros.

```
int main() {
    printf("Numero maximo de argumentos: %li \n",sysconf(_SC_ARG_MAX));
    printf("Numero maximo de hijos: %li \n",sysconf(_SC_CHILD_MAX));
    printf("Numero maximo de ficheros: %li \n",sysconf(_SC_OPEN_MAX));
    return 1;
}
```

```
gcc pr6.c -o pr6
```

```
./pr6
```

```
Numero maximo de argumentos: 2097152
```

```
Numero maximo de hijos: 62727
```

```
Numero maximo de ficheros: 1024
```

```
Variables _SC_ARG_MAX, _SC_CHILD_MAX, _SC_OPEN_MAX
```

Ejercicio 7. Escribir un programa que obtenga, con `pathconf(3)`, información de configuración del sistema de ficheros e imprima, por ejemplo, el número máximo de enlaces, el tamaño máximo de una ruta y el de un nombre de fichero.

```
int main() {
    char *path;
    printf("Numero maximo de enlaces: %li \n",pathconf("/", _PC_LINK_MAX));
    printf("Longitud maxima de ruta: %li \n",pathconf(".", _PC_PATH_MAX));
    printf("Longitudmaximadenombre de fichero: %li \n",pathconf(".", _PC_NAME_MAX));
    return 1;
}
```

```
gcc pr6.c -o pr6
```

```
./pr6
```

```
Numero maximo de enlaces: 65000
```

```
Longitud maxima de ruta: 4096
```

```
Longitud maxima de nombre de fichero: 255
```

```
Variables _PC_LINK_MAX_PC_PATH_MAX_PC_NAME_MAX
```

Información del usuario

Ejercicio 8. El comando `id(1)` muestra la información de usuario real y efectiva. Consultar la página de manual y comprobar su funcionamiento.

```
man 1 id
```

```
-a, -Z, -g, -G, -n, -r, -u, -z, -h, -v
```

Ejercicio 9. Escribir un programa que muestre, igual que `id`, el UID real y efectivo del usuario. ¿Cuándo podríamos asegurar que el fichero del programa tiene activado el bit *setuid*?

```
int main() {
    printf("UID: %i\n", getuid());
    printf("GUID: %i\n", getgid());
    printf("Usuario efectivo: %i\n", geteuid());
    printf("Grupo efectivo euid: %i\n", getegid());
return 1;
}
./pr9
UID: 1000
GUID: 1000
Usuario efectivo: 1000
Grupo efectivo euid: 1000
./pr10
Nombre usuario: cursoredes
Directorio home: /home/cursoredes
Desc usuario: cursoredes
```

Ejercicio 10. Modificar el programa anterior para que muestre además el nombre de usuario, el directorio *home* y la descripción del usuario.

```
#include <pwd.h>

int main() {
    struct passwd *s;
    if((s = getpwuid(getuid()))==NULL){
        perror("Error en el GETPWUID\n");
        return 1;
    }
    printf("Nombre usuario: %s\n", s->pw_name);
    printf("Directorio home: %s\n", s->pw_dir);
    printf("Desc usuario: %s\n", s->pw_gecos);
    return 0;
}

./pr10
Nombre usuario: cursoredes
Directorio home: /home/cursoredes
Desc usuario: cursoredes
```

Información horaria del sistema

Ejercicio 11. El comando `date(1)` muestra la hora del sistema. Consultar la página de manual y familiarizarse con los distintos formatos disponibles para mostrar la hora.

```
man 1 date
-d,-f,-I,-r,-R,--rfc,-s,-u,,--hrtp,--version
date +%d/%m/%y,%H:%M
28/10/21,19:12
```

Ejercicio 12. Escribir un programa que muestre la hora, en segundos desde el Epoch, usando la función `time(2)`.

```
#include <time.h>
int main() {
    time_t t = time(NULL);
    if(t == (time_t) -1){
        perror("Error TIME");
        return 1;
    }
    printf("Segundos desde el epoch: %lld\n", t);
    return 0;
}

./pr12
Segundos desde el epoch: 1635527793
```

Ejercicio 13. Escribir un programa que mida, en microsegundos usando la función `gettimeofday(2)`, lo que tarda un bucle que incrementa una variable un millón de veces.

```
man 2 gettimeofday
#include <sys/time.h>
int gettimeofday(struct timeval *tv, struct timezone *tz);
int settimeofday(const struct timeval *tv, const struct timezone *tz);
Feature Test Macro Requirements for glibc (see feature_test_macros(7)):
settimeofday(): _BSD_SOURCE

int main() {
    struct timeval ta;
    if(gettimeofday(&ta,NULL) == -1){
        perror("Error TIMEVAL");
        return 1;
    }
    int var =0;
    int i=0;
    for(i=0;i<1000000;++i){
        ++var;
    }
    struct timeval td;
    if(gettimeofday(&td,NULL) == -1){
        perror("Error TIMEVAL");
    }
}
```

```

        return 1;
    }
    printf("Tiempo para incrementar 1 millon: %lld\n", td.tv_sec-ta.tv_sec);
    printf("En microsegundos: %lld\n", td.tv_usec-ta.tv_usec);
    return 0;
}

```

./pr13

Tiempo para incrementar 1 millon: 0

En microsegundos: 2031

Ejercicio 14. Escribir un programa que muestre el año usando la función localtime(3).

```

man 3 localtime
#include <time.h>
struct tm *localtime(const time_t *timep);
struct tm *localtime_r(const time_t *timep, struct tm *result);

struct tm {
    int tm_sec;           /* seconds */
    int tm_min;           /* minutes */
    int tm_hour;          /* hours */
    int tm_mday;          /* day of the month */
    int tm_mon;           /* month */
    int tm_year;          /* year */
    int tm_wday;          /* day of the week */
    int tm_yday;          /* day in the year */
    int tm_isdst;         /* daylight saving time */
};

int main() {
    time_t t=time(NULL);
    if(t == (time_t) -1){
        perror("Error TIME");
        return 1;
    }
    printf("Año: %i\n", localtime(&t)->tm_year);
    return 0;
}

```

./pr14

Año: 121

```
printf("Año: %i\n", localtime(&t)->tm_year+1900);
```

./pr14

Ano: 2021

Ejercicio 15. Modificar el programa anterior para que imprima la hora de forma legible, como "lunes, 29 de octubre de 2018, 10:34", usando la función `strftime(3)`.

```
man 3 strftime
%A      The full weekday name according to the current locale.
%e      Like %d, the day of the month as a decimal number, but a leading zero is
replaced by a space. (SU)
%B      The full month name according to the current locale.
%Y      The year as a decimal number including the century.
%H      The hour as a decimal number using a 24-hour clock (range 00 to 23).
%M      The minute as a decimal number (range 00 to 59).

#include <locale.h>
int main() {
    time_t t=time(NULL);
    if(t == (time_t) -1){
        perror("Error TIME");
        return 1;
    }
    char s [100];
    setlocale(LC_TIME,"es_ES");
    strftime(s, 100, "%A, %e de %B de %Y, %H:%M\n", localtime(&t));
    printf(s);
    return 0;
}

./pr15
viernes, 29 de octubre de 2021, 19:45
```

Nota: Para establecer la configuración regional (*locale*, como idioma o formato de hora) en el programa según la configuración actual, usar la función `setlocale(3)`, por ejemplo, `setlocale(LC_ALL, "")`. Para cambiar la configuración regional, ejecutar, por ejemplo, `export LC_ALL="es_ES"`, o bien, `export LC_TIME="es_ES"`.